

an interface to a resource providing a service; and

means for receiving a first request from a client application coupled to the data

network, said first request containing an object-oriented, language independent, second request for access to the resource.—

--12. (Amended) A method for providing remote services by a server, coupled to a resource, comprising:

receiving a first request from a client application, said first request containing an

object-oriented, language-independent second request for access to the resource;

decoding the second request to determine a parameter needed by the resource; and

passing the parameter to the resource.--

REMARKS

The Examiner has taken the following position in the Advisory Action.

Applicant alleges that Luzeski does not teach using a language independent request for access the resources because JAVA is a platform independent programming language, not a language independent.

This is not found persuasive. Luzeski teaches using HTTP protocol for exchanging Java Applets or requests between client browser and server. Since HTTP is a language independent communications protocol, Luzeski's teachings are still seen meeting the claim limitation.

Applicant also alleges that since claim 1 contains means-plus-function language, COBRA (sic CORBA) software components should be read into the claim.

The examiner disagrees. First, it is submitted that COBRA (sic COBRA) was not explicitly cited in the present claims. Moreover, use of HTTP server was also disclosed in the present specification (see page 7, lines 16-17). Thus, reading the application of language independent HTTP protocol in to the present claims is indeed consistent with the present disclosure.

Essentially, if applicant is understanding the Examiner's position correctly, the Examiner appears to be taking the position that: a HTTP request containing a Java Applet is object

oriented (since the Java Applet is object oriented) and language independent (since HTTP is a language independent communications protocol). The Examiner appears to be taking the position, thus, that a HTTP server meets the limitation of a means for receiving a object oriented, language independent, request.

Applicant has amended claim 1 to make it clear that the means for receiving claimed in this application is not simply an HTTP server. Specifically, claim 1 has been amended to recite “means for receiving a first request from a client application coupled to the data network, said first request containing an object-oriented, language independent, second request for access to the resource.”

The first part of this claim amendment recites “means for receiving a first request from a client application coupled to the data network.” This first request is the protocol request, such as a GIOP² request, an IIOP³ request or an HTTP request, although the invention is not limited to a means for receiving first requests formatted according to GIOP, IIOP, or HTTP protocols.

The second part of this claim amendment recites “said first request containing an object-oriented, language independent, second request for access to the resource.” This second part makes clear that the protocol request contains another request that is both (1) object oriented and (2) language independent.

Since a Java Applet is object oriented, but not language independent, including a Java Applet (second request) in a HTTP request (first request) does not anticipate amended claim 1. Accordingly, applicant respectfully requests that the Examiner withdraw the rejections of claims 1-11 over Luzeski, alone or in combination with Osder.

² General Inter-ORB Protocol

³ Internet Inter-Orb Protocol

Independent claim 12 has been amended to recite that the method includes “receiving a first request from a client application, said first request containing an object-oriented, language-independent second request for access to the resource.” Thus, claim 12 is patentable for the same reasons set forth above with respect to claim 1. Accordingly, applicants respectfully request that the Examiner withdraw the rejection of claim 12 over Luzeski.

Request for Interview

If the Examiner believes that continued rejection of the claims is warranted, after reviewing the arguments presented herein and the arguments presented in the Reply dated October 8, 2002, applicant respectfully requests that the Examiner contact the applicant’s representative at the number below to schedule a telephone interview to discuss this application. If possible, applicants would like to conduct the interview prior to entry of the next Official Action.

Conclusion

In view of foregoing claim amendments and remarks, it is respectfully submitted that the application is now in condition for allowance and an action to this effect is respectfully requested. If there are any questions or concerns regarding the amendments or these remarks, the Examiner is requested to telephone the undersigned at the telephone number listed below.

Amendment Dated December 16, 2002
Serial No. 09/223,972

If any fees are due in connection with this filing, the Commissioner is hereby authorized to charge payment of the fees associated with this communication or credit any overpayment to Deposit Account No. 502246 (Ref: NN-HU0125).

Respectfully Submitted


John C. Gorecki
Registration No. 38,471

Dated: December 16, 2002

John C. Gorecki, Esq.
Patent Attorney
165 Harvard St.
Newton, MA 02460
Tel: (617) 796-9024
Fax: (617) 795-0888

VERSION WITH MARKINGS TO SHOW CHANGES MADE

Submitted herewith is a marked-up version of the amended claims to show changes made in the foregoing Amendment.

IN THE CLAIMS

Claims 1 and 12 have been amended as follows:

--1. (Amended) A server comprising:

an interface to a data network

an interface to a resource providing a service; and

means for receiving a first request from a client application[,] coupled to the data network, said first request containing an object-oriented, language independent, second request for access to the resource.--

--12. (Amended) A method for providing remote services by a server, coupled to

a resource, comprising:

receiving a first request from a client application, said first request containing an object-oriented, language-independent second request for access to the resource; decoding the second request to determine a parameter needed by the resource; and passing the parameter to the resource.--